

1 METHOD AND APPARATUS FOR DETECTING A  
2 STATIONARY DITHER CODE  
3

4 CROSS REFERENCE TO RELATED APPLICATION

5 This application claims priority from U.S. Provisional Patent Application Serial  
6 Nos. 60/164,946, entitled "Stationary Dither Code Detection," 60/164,952, entitled  
7 "Optimal Stationary Dither Code," and 60/164,951 entitled "Non-stationary Dither Code  
8 Detection," all filed November 12, 1999. The disclosure of these provisional patent  
9 applications are incorporated herein by reference in its entirety.  
10

11 BACKGROUND OF THE INVENTION

12 Field of the Invention

13 The present invention relates to data communication systems and, more  
14 particularly, to data communication systems using codes with coding boundaries that vary  
15 according to a dithering pattern.  
16

17 Description of the Related Art

18 Pseudonoise (PN) codes, also known as pseudorandom codes, have good  
19 autocorrelation properties and are ideal for measuring the time-of-arrival of a signal, and  
20 hence, for use in ranging applications. A high coding gain is achieved when a receiver  
21 correlates the code, such as a PN code, for a long time. Accordingly, long PN codes have  
22 been used to provide high coding gain.

23 An example of using a long PN code is in the Global Positioning System (GPS).  
24 GPS is a satellite-based navigation system in which each satellite in the GPS constellation  
25 transmits a unique precision (P) code. Each P code is a very long PN code that has a  
26 period of one week, and hence, repeats only once per week. The P code provides for a  
27 high gain correlation signal that allows a receiver to acquire precise timing information,  
28 and hence, precise ranging information between the receiver and satellite. To acquire the  
29 P code signal a receiver must determine the phase of the received P code. However,  
30 because the P code is so long it is difficult for a receiver to determine its phase and

1 therefore is difficult to acquire. If a receiver includes a very accurate clock, such as an  
2 atomic clock, it might stay synchronized with the transmitter's clock and could therefore  
3 generate a local reference P code that is nearly in-phase with the received P code. But  
4 this would require an expensive, complex and unwieldy receiver. Even with a very  
5 accurate and stable clock frequency, there is the problem of initially setting the correct  
6 time of day. This requires testing all of the code phases that span the range of the time  
7 uncertainty. Theoretically a receiver could include enough correlators to test each phase  
8 of the P code simultaneously to detect its phase, but in practice that solution would not  
9 be feasible because of the extremely large number of correlators required due to the  
10 length of the P code. A receiver also, theoretically, could use a practical number of  
11 correlators and test the possible phases of the P code sequentially, however, that would  
12 take a long time due to the length of the P code.

13 The GPS system uses another technique to acquire the P code. It uses a second  
14 signal, known as the course acquisition (C/A) signal, to determine initial timing  
15 information, then it uses that timing information to assist in detecting the phase of the P  
16 code. The C/A signal is a much shorter code of 1023 symbols and repeats approximately  
17 every millisecond. Accordingly, the C/A code can be detected quickly, either by using  
18 a bank of correlators to test each possible phase of the code simultaneously, or by testing  
19 the phases in sequence. However, the C/A code can provide timing only within one  
20 millisecond interval, because one such interval cannot be distinguished from another, and  
21 thus, by itself, cannot resolve all of the time uncertainty, which typically is a few seconds.  
22 But once a GPS receiver acquires the C/A signal it can determine enough timing  
23 information to limit the P code search to specific phases of the P code to speed its  
24 acquisition of the P code. However, using an ancillary code to help acquire a longer  
25 code, such as in the GPS system, requires that the transmitter generate and transmit two  
26 signals and that a receiver be able to receive and detect both signals.

27 Accordingly, there is a need to generate, transmit and receive a long code with  
28 good autocorrelation characteristics that can be acquired quickly using a feasible number  
29 of correlators and without requiring another signal that provides timing information.

30 A conventional spread-spectrum communications system, such as the GPS  
31 system, is shown in Fig. 1. The system of Fig. 1 includes a transmitter 1 and a receiver

1 9. The transmitter includes a PN code generator 2 that is controlled by a timing counter  
2 3 and both are driven based on a clock oscillator 4. The PN code generated by generator  
3 2 is modulated with a carrier signal via modulator 5 which is driven by carrier oscillator  
4 6. Optionally, data can be superimposed onto the code and carrier by using a modulo 2  
5 adder 7. The transmitter 1 transmits the modulated carrier via antenna 8 to a receiver 9.

6 Receiver 9 receives the transmitted signal via an antenna 10 that provides the  
7 received signal to a demodulator 11. The demodulator 11 is driven by a carrier oscillator  
8 12, and produces two signals out-of-phase by  $90^\circ$ . Those signals are designated as in-  
9 phase (I) and quadrature(Q) signals. These two out-of-phase signals are provided to a  
10 group of parallel correlators 13. The parallel correlators can include as many, and even  
11 more, correlators as the number of phases of the code to be tested. For example, if the  
12 code length is 1023 symbols, or chips, the parallel correlators 13 can consist of 1023  
13 correlators, with one correlator for each possible phase of the code. Multiple banks of  
14 the parallel correlators 13 can be used to correlate different signals, such as in this case  
15 where one bank correlates the I-signals and another bank correlates the Q-signals. The  
16 parallel correlators 13 are also provided with PN reference codes that correspond to the  
17 PN codes generated in the transmitter. PN code generator 14 generates the reference  
18 codes. The reference codes can be delayed to correspond to the various phases to be  
19 tested. Alternatively, the input signals, here the I and Q signals, can be delayed with  
20 various delays and correlated with a single PN code to test the different phases. The PN  
21 code generator 14 is driven by a local clock oscillator 15 and timing counters 16 which  
22 can produce the different timings for the PN reference codes. The local clock oscillator  
23 also drives timing counters 16.

24 Many different types of PN code generators can be used in spread-spectrum  
25 systems such as that shown in Fig. 1. One such conventional, simplified PN code  
26 generator is shown in detail in Figure 2. The PN code generator of Fig. 2 includes a shift  
27 register 20 and a modulo-2 adder 21. In this example, the shift register is six bits long,  
28 although it can be longer or shorter as needed. The six flip-flops of shift register 20 are  
29 synchronously driven by a clock signal. Modulo-2 adder 21 adds the first and sixth bits  
30 of the shift register, and inserts the result into the first position of register 20. In this  
31 manner, a maximal-length PN sequence is produced.

1           The GPS receiver 9, shown in Fig. 1, requires two PN code generators. One of  
2 the PN generators, similar in concept to the generator shown in Fig. 2, generates the P  
3 code and another, also similar to that shown in Fig. 2, generates the C/A code. Although  
4 only one generator is shown in the Fig. 1 it will be understood that in the case of a long  
5 code system, such as GPS, two generators are used. Also, the receiver, in the threshold  
6 detection function 17, must be able to detect both the P and C/A codes. Accordingly,  
7 conventional spread-spectrum receivers, such as GPS receivers, suffer from the problems  
8 described above and are unable to detect a long code quickly and inexpensively without  
9 the aid of a second timing signal. Accordingly, there is a need for a receiver that can  
10 detect a long code with good autocorrelation properties, yet can acquire it quickly and  
11 inexpensively without the need for first detecting a second timing signal.

#### 12 13                                   SUMMARY OF THE INVENTION

14           Therefore, in light of the above, and for other reasons that become apparent when  
15 the invention is fully described, an object of the present invention is to detect a long code  
16 from a sequence of shorter codes, where the shorter codes are dithered versions of the  
17 same short code.

18           A further object of the present invention is to use a single pseudonoise reference  
19 code generator to generate a single reference code to acquire a long code composed of a  
20 sequence of dithered short codes.

21           Yet a further object of the present invention is to detect a long code having a  
22 predetermined dither pattern.

23           Another object of the present invention is to determine a maximum correlation  
24 value from a plurality of correlation signals produced by a plurality of correlators.

25           Yet another object of the invention is to detect a sequence of dithered short code  
26 composing a long code.

27           A further object of the invention is arrange correlation results representing  
28 candidate matches of a long code so the most likely candidate for the match is readily  
29 available for a access by a processing unit.

30           A still further object of the present invention is to provide an apparatus that  
31 achieves one or more of the above objects.

1           The aforesaid objects are achieved individually and in combination, and it is not  
2 intended that the present invention be construed as requiring two or more of the objects  
3 to be combined unless expressly required by the claims attached hereto.

4           In accordance with one aspect of the present invention, there is provided a method  
5 of detecting a transmission code from a received signal, where the transmission code is  
6 composed of a plurality of dithered codes. The method includes detecting the plurality  
7 of dithered codes, and detecting the long code based on the detected dithered codes. The  
8 method generates a detection signal for each detected dithered code, combines the  
9 detection signals, and detects the long code based on the combination of detection  
10 signals. In the case where the composite code includes M dithered codes, the correlation  
11 signals can be combined by summing the M correlation sums to generate a present final  
12 sum. The long code can be detected by determining a largest final sum from among the  
13 present final sum and previously generated final sums where one of the sums will be large  
14 because those M correlation sums align with the M dither codes in the received signals.  
15 By associating a time of the largest final sum with the time of the transmission code, the  
16 timing of the long code is determined. The dither pattern can be a stationary dither  
17 pattern, that is fixed and well-known.

18           According to another aspect of the invention there is provided a dither code  
19 detector that receives a coded signal having a long code composed of a plurality of  
20 dithered codes, where the dithered codes are dithered according to a dither pattern. The  
21 detector includes a correlator unit that correlates a coded signal with a reference code and  
22 outputs a correlation signal. It also includes a detector that combines portions of the  
23 correlation signal, corresponding to the dithered codes, and detects the long code based  
24 on the combined portions of the correlation signal. The detector can include a delay unit,  
25 such as a series of RAMs, that receives the correlation signals and delays them according  
26 to the dither codes. It can also include a combiner that receives the present correlation  
27 signal and combines it with the delayed correlation signals, to generate a combined  
28 correlation signal. This delaying and combining is performed for each delay code in the  
29 composite code, and the final combiner outputs a final correlation sum that represents the  
30 sum of the correlations of the individual dither codes over the entire period of the long  
31 code. When the correlation for the last dithered code is input to the delay unit, and hence,

1 the end of the long code enters the delay unit, then the detector will output a strong final  
2 correlation sum near the end of the long code. Time information is associated with the  
3 correlation signals, and when the strong final combined correlation occurs, the time of  
4 the beginning of the long code can be determined from the time information. The RAMs  
5 can delay the correlation sums by being addressing with a modulo counter using a  
6 modulus corresponding to the amount of dither being detected.

7 The above and still further objects, features and advantages of the present  
8 invention will become apparent upon consideration of the following descriptions and  
9 descriptive figures of specific embodiments. While these descriptions go into specific  
10 details of the invention, it should be understood that variations may and do exist and  
11 would be apparent to those skilled in the art based on the descriptions herein.  
12

#### 13 BRIEF DESCRIPTION OF THE DRAWINGS

14 Fig. 1 is a block diagram of a conventional spread-spectrum communication  
15 system.

16 Fig. 2 shows a conventional pseudonoise generator.

17 Fig. 3A shows an example boundary dither code.

18 Fig. 3B shows an example rotation dither code.

19 Fig. 3C shows alignments of a reference code with the boundary dither and  
20 rotation dither codes shown in Figs. 3A and 3B that a receiver might use to detect either  
21 of the dither codes shown in Figs. 3A and 3B.

22 Fig. 4 is a block diagram of a receiver according to one embodiment of the  
23 invention.

24 Fig. 5A is a block diagram of the maximum likelihood decoder shown in Fig. 4.

25 Fig. 5B is a block diagram of the maximum likelihood decoder shown in Fig. 4,  
26 according to another embodiment of the invention.

27 Figs. 6A-D are code timing diagrams for showing the operation of the maximum  
28 likelihood decoder shown in Fig. 5.

29 Figs. 7A and B are flowcharts illustrating a method of detecting a stationary dither  
30 code according to the invention.

31 Fig. 8 is a block diagram of the maximum finder shown in Figs. 4 and 5.

1 Fig. 9 is a conceptual diagram for showing the operation of the maximum finder  
2 shown in Fig. 8.

### 3 4 DESCRIPTION OF THE PREFERRED EMBODIMENTS

5 Preferred embodiments according to the present invention are described below  
6 with reference to the above drawings, in which like reference numerals designate like  
7 components.

#### 8 *Overview*

9 A co-pending patent application, entitled "Method and Apparatus for Generating  
10 and Transmitting a Stationary Dither Code," relates to generating and transmitting  
11 stationary dither codes, including an optimal dither code, and is incorporated herein by  
12 reference. In that co-pending patent application the generation and transmission of  
13 stationary dither codes and optimal stationary dither codes are described, as is the code  
14 structure that typically would be used in a data communications transmitter. The present  
15 invention is directed to methods and apparatuses for detecting any stationary dither code,  
16 both optimal and non-optimal stationary dither codes. Such methods and apparatuses  
17 typically would be used in a data communications receiver. In the following description  
18 of the invention a simple, non-optimal code is used for purposes of illustrating the  
19 invention. However, the invention is not limited to such a code, and applies to detecting  
20 any stationary dither code.

#### 21 *Dither Types*

22 A dither code can be generated from a predetermined code, by changing certain  
23 aspects of that predetermined code. There are many PN codes that can be used, such as  
24 m-sequences produced by a generator such as that shown in Fig. 2. For example, a  
25 reference code N chips, or bits, long with good autocorrelation properties is selected. A  
26 code generator can generate a long code by repeating the reference code M times.  
27 However, so that a receiver can distinguish one part of the long code from another, the  
28 generator varies the timing, or dithers, the repetitions of the reference code. The varied  
29 timing repetitions of the reference code are referred to here as short codes. The original  
30 undithered code is referred to here as the reference code, because a receiver typically will  
31 generate the reference code and use it as a reference to detect the dithered short codes.

1           One kind of dither is a boundary dither that is created by varying the lengths of  
2 the short codes by small amounts. All of the short codes start the same way, but a  
3 generator adds or deletes bits at the ends of the codes in order to vary the length of the  
4 reference code. The boundaries between the short codes are thus not equally-spaced, but  
5 are varied by the dither pattern. The length of the long code is nominally  $M \cdot N$  chips, but  
6 perhaps not exactly that length, because of the length variations due to the boundary  
7 dither.

8           Another kind of dither is rotation dither, performed by rotating the short codes, yet  
9 keeping the code length constant. That is, the short codes start in varying states, or at  
10 varying phases of the code, and each rotation dithered short code includes the entire  
11 reference code. The boundaries between the rotation dithered short codes are equally-  
12 spaced since the code lengths are the same, but there remains one feature in common with  
13 the boundary dither. Namely, if a dithered code is compared with an undithered code, the  
14 code phase of the short codes will appear to shift back and forth in some pattern referred  
15 to here as the dither pattern. The length of the long code is exactly  $M \cdot N$  symbols, or  
16 chips, in the case of rotation dither.

17           Here, the dither patterns, also called dither codes, are fixed, and thus they are  
18 referred to as stationary dither codes. Here too, the long code has a constant length in  
19 each repetition of the long code.

20           Figs. 3A-C illustrate the two types of dither with a small-scale example, in which  
21 the 7-bit code "0010111" is repeated four times. Dithering that 7-bit reference code by  
22 the boundary dither method varies the lengths of the codes. The codes marked in Fig. 3A  
23 as (-1) are one bit shorter, and those marked (+1) are one bit longer than the reference  
24 code.

25           Dithering the 7-bit reference code by the rotation dither method shifts the codes.  
26 The codes marked in Fig. 3B as (-1) are rotated one bit to the left, those marked (+1) are  
27 rotated one bit to the right, and those marked (0) are not rotated.

28           For either type of dither, a receiver can generate a reference code (an unmodified  
29 7-bit code) and find phase shifts of the reference code that are aligned with the received  
30 codes, as illustrated in Fig. 3C. The aligned reference codes and received codes will  
31 match almost entirely, except for a few bits at the ends of the codes, and thus, will

1 produce a strong correlation. Accordingly, a code with a long period can be generated  
2 using only a single, short reference code.

### 3 *Short Code Detection*

4 A receiver according to one embodiment of the invention is shown in Fig. 4.  
5 Here, many of the front-end components of the conventional receiver 9 shown in Fig. 2  
6 are used, up through the parallel correlators 13, although only one reference PN code  
7 generator 14 is needed. However, the threshold detection functional unit 17 of the  
8 conventional receiver is replaced with a maximum likelihood decoder 40 and a  
9 time/range detector 42, as shown in Fig. 4.

10 The dithered short codes, composing the received long code, are detected by a set  
11 of correlators 13 that correlate the received signal with various time delays of an  
12 undithered reference code. The undithered reference code has N symbols and is produced  
13 by PN code generator 14. Here, the correlator unit 13 includes two banks of correlators  
14 to process the I and Q signals, and each bank includes N correlators, using delays of 0,  
15 1, 2, .. N-2, N-1 symbols, or chips. Generally, correlators are provided at half-chip  
16 spacing, but for ease of explanation and understanding, the correlator unit 13 provides  
17 one correlator per symbol in the undithered code. The correlators are dumped  
18 sequentially and cyclically; that is, the correlator units dump them in the sequence 0, 1,  
19 2, .. N-2, N-1, 0, 1, 2, .. (using the delay values to identify the correlators). In each code  
20 interval, one of the correlators will be best aligned with the received code, and will match  
21 most of the N chips of the reference code. More precisely, either the received code will  
22 be exactly aligned with one of the correlators, giving it maximum detection, or the  
23 received code will be aligned halfway between two adjacent correlators, giving each half  
24 the maximum, or some combination between these extremes.

25 Generally, not all of the N chips of the reference code are matched because the  
26 short codes in the received signal are dithered. Accordingly, there is loss of gain due to  
27 all of the dithered signal not being correlated with the receiver's reference code. In order  
28 to minimize this loss of gain a transmitter produces the short codes by dithering them  
29 only by small amounts. Both boundary and rotation types of dither have this small loss  
30 of gain characteristic. In the case of boundary dither, the unmatched bits occur at the end

1 of the short codes; and in case of rotation dither, the unmatched bits occur at both ends  
2 of the short codes.

3 The parallel correlator unit 13 output correlation sums. The sum is large when  
4 the received signal correlates strongly with the reference code. Accordingly, the  
5 correlators detect the short codes when the correlation is strong and the correlation sum  
6 output is large.

### 7 *Dither Code Detection*

8 Because the dither code is fixed, or stationary, the dither pattern is known ahead  
9 of time and the receiver can be designed to detect it. Thus, a receiver designed for a  
10 specific dither pattern used by a transmitter can detect the dither pattern by summing  
11 appropriate sequences of the correlator outputs and finding the highest total correlation.  
12 A high total correlation summed over sequences of the correlator outputs corresponding  
13 to the dither pattern indicates that the receiver has detected the phase of the dither pattern,  
14 and thus, that it has detected the phase of the long code, that is, the time of arrival of the  
15 long code. Once the receiver detects the phase of the long code it can process the long  
16 code such as by determining timing information and/or range information based on the  
17 long code.

18 The receiver 9 shown in Fig. 4 detects the dither code using a maximum-  
19 likelihood decoder 40. The maximum likelihood decoder 41 decodes the received signal  
20 by using a matched filter 41 and detects the dither code (for either type of dither) by using  
21 a maximum finder 42 that processes the output of the matched filter 31. The matched  
22 filter sums sequences of correlation signals output from the parallel correlators 13 over  
23 a period time corresponding to the period of the dither code. The maximum finder 42  
24 arranges the sums output from the matched filter to store them in order of the strength of  
25 the sums, where the strongest sum corresponds to the best indication of the phase of the  
26 dither code. The sums arranged by the maximum finder 42 are made available for  
27 processing by time/range detector 43. By arranging the sums in strength order, the most  
28 appropriate sums, i.e., those most likely to correspond to the long code, are readily  
29 available to be provided quickly to a processing unit. The maximum likelihood decoder  
30 40 will now be described in detail.

### Maximum-Likelihood Decoder

One embodiment of the maximum likelihood decoder 40 is shown in Fig. 5A, and includes a matched filter 41, matched to the signal structure, receiving the output from parallel correlators 13 and outputting a sum "A" to maximum finder 42.

The correlator output comes from a set of N correlators that correlate the received signal with an undithered reference signal delayed 0, 1, 2, .. N-2, and N-1 chips. Because the correlators are dumped sequentially and cyclically, this output is a sequence of correlation sums output at the chip rate in the order of the delay values. Those correlation sums are input to the matched filter 41.

Matched filter 41 includes a series of delay units, in this case random access memories (RAMs), counters and adders. For a long code composed of L short codes, the decoder needs L-1 RAMs and L-1 adders as shown in Fig. 5A. Accordingly, the matched filter 41 includes (RAMs)  $50_a$  through  $50_{L-1}$ , modulo counters  $51_a$  through  $51_{L-1}$ , and adders  $52_a$  through  $52_{L-1}$ . The counters count modulo N1, N2, N3, etc. as shown in the figure, where each modulus is set equal to the length of one of the short codes in the case of boundary dither, or equal to the short code length plus the equivalent phase shift values in the case of rotation dither. The matched filter also includes modulo N counter 53 and modulo L counter 54 which has a counting rate equal to the cycling rate of modulo N counter 53. The chip-rate clock is supplied to each of the modulo counters  $51_a$  through  $51_{L-1}$ , and to modulo N counter 53 as shown in Fig. 5A, and controls the counting rate of those counters. The sequence of values provided by the series of delay units matches the dither pattern dither code.

The RAMs provide delay-line memory, and each RAM is addressed by a counter with a modulus equal to one of the dither code values N1, N2, N3, etc. The modulus value of each counter sets the data delay through the corresponding RAM. On each clock cycle, the addressed word is read (output) from the RAM and the same memory location is replaced by writing the input by using, for example, a read-modify-write cycle. For a counter modulus of, for example, N-1, if a correlator sum is written in clock cycle n, it will be read in clock cycle n+N-1, that is, after a delay of N-1 clock cycles. These counters do not need to be synchronized to any other timing to provide these delays. If

two or more RAMs need the same delay, they may share the same counter, thus reducing the hardware cost.

### *Dither Code Correlation*

The adders shown in Fig. 5A produce a sum of correlator sums that belong to the same dither code. This is best explained with an example using a boundary dither code with a small value of  $L$ . In this example there are ten (10) short codes in the long code, and hence  $L=10$ . Accordingly, the matched filter will include 9 RAM/adder units. The dither code values in this example are +1, -1, -1, +1, -1, +1, +1, -1, +1, -1. Accordingly, the lengths of the short codes will be  $N+1$ ,  $N-1$ ,  $N-1$ ,  $N+1$ ,  $N-1$ ,  $N+1$ ,  $N+1$ ,  $N-1$ ,  $N+1$ ,  $N-1$ , respectively. The RAMs will therefore be connected to counters with moduli that match the first nine (9) of these lengths, in that order.

Figs. 6A through 6D are timing diagram that illustrate how this code is detected. Fig. 6A shows the 10 short codes of the received dither code, and their lengths. It happens that in this example, just three of the correlators are involved in detecting this dither code.

Fig. 6B shows the staggered timing of outputs of these correlators, and which of the short codes are detected by which of the correlators. Note that in using a boundary dither, in this example, the start of the correlation intervals align with the short codes, but not the ends of the intervals.

Next, the top part of Fig. 6C shows the cycles of the counter with modulus  $N-1$ , and the bottom part of that figure shows the delays provided by the RAMs addressed by the counters with that modulus. Fig. 6D shows the cycles of the counters with modulus  $N+1$ , and the delays provided by RAMs addressed by those counters. For example, the delay of the sum of correlations for short codes 1 and 2 (shown in the bottom of Fig. 6C) begins when the delay for sum 1 ends (shown in the bottom of Fig. 6D) and also when the correlation for short code 2 ends (shown in Fig. 6B). Because the counter counts modulo  $N+1$  the delay of short code 1 ends at the same part of the counter cycle as it began, and likewise with the  $N-1$  counters.

A process for detecting a long code composed of a sequence of shorter dithered codes is shown in Fig. 7A. The process starts at operation 70, and in operation 71 parallel correlators 13 correlate the received signal with delay reference codes provided

by PN code generator 14, and they dump correlation sums to the maximum likelihood detector 40. The maximum likelihood detector's matched filter 41, in operation 72, combines the correlation signals according to a predetermined dither pattern identical to the dither pattern used by the transmitter in generating the transmitted long code, and outputs the combined correlation signals. Maximum finder 41, in operation 73, arranges the combined correlation signals according to their strength, with the strongest correlations being stored in a first location. The time/range detector 43, in operation 74, determines the phase of the long code based on the strength of the correlation signals stored in the maximum finder 42 and the receive times of those correlation signals. In operation 75 the time/range detector 43 processes the correlation signals output from the maximum finder 42 based on the detected phase of the long code, to determine timing information or ranging information from the long code.

Next, a more detailed description of operation 72 and the matched filter 41 is provided with reference to Figs. 5A and 7B.

N chips after the received dither code begins, one of the correlators, for instance, correlator  $i$ , which detected the first short code  $C_0$ , dumps its result to maximum likelihood detector 40, even though it did not examine the last chip of this code, as indicated in operation 76. The sum from correlator  $i$  is delayed by storing it in the first RAM  $50_a$  (addressed with modulus  $N+1$ ) at the location given by the current state of counter  $51_a$ ,  $x$ , in operation 77.

One chip later, the second received short code  $C_1$  begins.

N chips later, the second received short code  $C_1$  is detected by correlator  $i+1$ , because the time is  $N+1$  chips after the correlator  $i$  detection. The correlation result is strong even though the correlator has tried to examine one chip too many. The sum from correlator  $i+1$  arrives at the first adder  $52_a$ . Since the time is now  $N+1$  cycles after the first sum was stored in the first RAM  $50_a$ , the address counter for the first RAM has advanced to  $x+N+1$  modulo  $N+1$ , which equals  $x$ , so that the correlator sum for the first short code is now read from location  $x$  of RAM  $50_a$ , and also arrives at the first adder  $52_a$  at the same time as the second short code  $C_1$ . This adder, in operation 78, adds the sums for the first two short codes, and in operation 79 stores the result the second RAM  $50_b$  (with modulus  $N+1$ ) at the current state of its counter,  $y$ .

1           One chip earlier, the second of the received short codes had ended, and the third  
2 short code  $C_2$  begins here.

3           N chips later, the third received short code  $C_2$  is detected by correlator i, because  
4 the time is N-1 chips after the correlator i+1 detection. Again, the correlation result is  
5 strong even though the correlator has tried to examine one chip too many. The sum from  
6 correlator i (for the third short code) arrives at the second adder. Since the time is now  
7 N-1 cycles after the second sum was stored in the second RAM, thereby delaying the  
8 second sum as indicated in operation 80, the address counter for the second RAM has  
9 advanced to  $y+N-1$  modulo N-1, which equals y, so that the second sum is now read from  
10 RAM 50<sub>b</sub>, and also arrives at the second adder 52<sub>b</sub>. This adder, in operation 81, adds the  
11 correlator output for the third short code  $C_2$  to the second sum to produce the third sum.  
12 Because there are still more codes to process in the dither pattern, as determined in  
13 operation 82, flow returns to operation 79 where the third sum is stored in the third RAM  
14 50<sub>c</sub>.

15           This process continues until the sum of the correlations for the first nine short  
16 codes is read from the ninth RAM 50<sub>L-1</sub>, just as the correlation for the tenth short code  
17  $C_9$  is dumped. These two values are added by the last adder 52<sub>L-1</sub> to obtain the total  
18 correlation for the dither code. Since the tenth short code is the last code in the dither  
19 pattern, as determined in operation 82, flow goes to operation 83, in which the final  
20 correlation sum is output to the maximum finder 42. Note that this time, when the final  
21 correlation sum is output from adder 52<sub>L-1</sub>, is not aligned with the end of the dither code.  
22 Rather, this time is a fixed delay (i.e.,  $N*L+1$  chips) from the start of the dither code and  
23 may not perfectly align with the end of the dither code because the varying length of the  
24 short codes while the correlators operate on a fixed length basis. That property is  
25 exhibited here, where the end of the dither code is not yet received at the time when adder  
26 52<sub>L-1</sub> outputs the final sum to the maximum finder.

27           Note that the final sums produced at times other than the time corresponding to  
28 the end of the long code will generally reflect the summing of a haphazard assortment of  
29 unaligned data that generally will produce small, weak values. Note also that other dither  
30 codes may arrive with other timing such that other correlators and other RAM words are

involved, but the time of arrival of the final sum will always be the same delay from the start of the dither code.

The modulo-N counters and the modulo-L counter discussed below, shown in Fig. 5 are part of a chain of counters that define the local timing (uncorrected clock) of the receiver. The maximum finder uses these counters to record the time-of-arrival of the final sums. The maximum finder logic sorts the data as it arrives, keeping the data for the K highest correlations. K is a hardware cost parameter that will depend on the application. For some applications, K=1 may suffice. Basically, the maximum finder sifts out the best (or most promising) data for further analysis.

An alternate embodiment of the maximum finder and matched filter is shown in Fig. 5B. Here, the same components are used as in the matched filter shown in Fig. 5A, but they are arranged differently. In comparison to the matched filter in Fig. 5A, the adders in Fig. 5B are removed from between the RAMs, and the string of RAMs/counter pairs is reversed. Accordingly, the modulo N1 counter is placed at the output end of the filter, as shown, and the modulo N(L-1) counter is placed at the input end of the filter. The string of adders  $52_{L-1}$  to  $52_a$  receive as inputs the outputs of the respective RAMs and the output of the preceding adder. The last adder in the filter, adder 52a, supplies the output A to the maximum finder 42. The matched filter shown in Fig. 5B is an example of a matched filter that is a mathematically equivalent form of the matched filter shown in Fig. 5A, and can be used in place of the matched filter of Fig. 5A.

It will be understood that the matched filter according the present invention can be implemented in hardware, software or firmware.

#### *Maximum Finder*

If the application of the dither code detector is such that there is just one long code, as, for example, to mark the beginning of a message), then the processing needed for the output of the matched filter ("A" in Figs. 5A and B) is as follows: Compare the output to a threshold value, and when the output exceeds the threshold, the long code will generally be detected. At this time, the outputs B and C of modulo N counter 53 and modulo L counter 54 will indicate the time of arrival of the long code, and this information can be used for ranging estimates, or message synchronization, or otherwise as useful to the application. The setting of the threshold will generally be important when

1 noise is present at the receiver's input. If the threshold is too low, the detection decision  
2 can be accidentally triggered by the noise; but if the threshold is too high, there may be no  
3 detection at all.

4 In other applications, the long code may be repeated, providing multiple  
5 opportunities for detection. Generally, these applications will be able to tolerate higher  
6 levels of noise than the applications described above. By combining the data from  
7 multiple long code detections, it is possible to get sufficiently reliable timing information  
8 when higher levels of noise are present at the receiver's input. However, under such high  
9 noise conditions, there will be many false (unaligned) correlations caused by the noise  
10 that are as strong as the correct (aligned) correlations.

11 It is common practice to use a computer and software, such as with the processing  
12 unit described below, to process correlation sums and associated timing information. But  
13 in the case of multiple code detections, the computer and software may not be able to  
14 keep pace with the data rate, in which case some data will be lost. If some data must be  
15 dropped, it will be best to retain the best data, that is, the strongest correlations, because  
16 these are less likely to be false correlations caused by noise. The purpose of the  
17 maximum finder described next is to retain the best data in such a situation.

18 The maximum finder 42, also called a ranking list, sorts the sums as they are  
19 produced, placing the strongest sums in a list. This list stores the amplitudes of the final  
20 correlator sums, representing the strength of the correlations, and the received times of  
21 those correlations. The received time, also referred to as a time tag, represents the arrival  
22 time of the code boundary in terms of the receiver's local timing. The received time is  
23 derived from the receiver's local time using counters 53 and 54, when the correlation sum  
24 is transferred from the correlators to the maximum finder, which is typically one  
25 reference code length (nominal code length) after the beginning of the received code.  
26 Because the actual code length varies, the end of the received dither code is typically  
27 close to, but usually not exactly, when the received code ends.

28 When the list is full, correlations weaker than the weakest correlation in the list  
29 are discarded; and when a new correlation is added to the list, the weakest correlation in  
30 the list is removed to make room. The maximum finder is structured to keep the  
31 strongest correlation at the top of the list so it is readily available for a processing unit to

1 access it. A processing unit, such as time/range detector 43, takes the strongest  
2 correlation from the maximum finder to process it, thereby removing it from the list.  
3 Keeping the correlation sums sorted so the maximum correlation sum is stored at the top  
4 of the list and readily available increases the chances that aligned (i.e., correct) correlation  
5 sums will not be missed. The size of the list, corresponding to the number of registers  
6 in the maximum finder, is generally set equal to the average number of correlations that  
7 the processing unit can process per code period. Approximately, a smaller size  
8 diminishes performance (e.g., the speed and reliability of detection), but a larger size  
9 merely wastes hardware resources. The faster the processing unit, the more correlation  
10 sums should be stored in the maximum finder to increase system performance.

11 The maximum finder 42, according to one embodiment of the invention, uses the  
12 repetitive structure as shown in Fig. 8 to perform insertion sorting of the final correlation  
13 sums output from the matched filter 41. In Fig. 8, input "A" is the amplitude of the final  
14 correlation signal output from the matched filter, and inputs "B" and "C" are time-of-  
15 arrival counts corresponding to the final sums. The input B is generated by modulo N  
16 counter 53 and input C is generated by modulo L counter 54, both shown in Figs. 5A and  
17 B. The maximum finder includes a plurality of registers, such as registers 100a through  
18 100c shown in Fig. 8. Each register holds the values of A, B and C, namely, a final sum  
19 amplitude A, a count B from modulo N counter 53, and a count C from modulo L counter  
20 54. The registers are arranged in Fig. 8 to show the register 100a that holds the  
21 maximum sum amplitude at the top of the figure, followed by the register 100b that holds  
22 the next lower amplitude value, or "runner-up" amplitude, arranged below register 100a,  
23 and the register 100c holding the next smaller amplitude value, or the "third-place"  
24 amplitude, arranged below the runner-up register. Further registers would be arranged  
25 in a continuing similar order below the third-place register, and would hold increasingly  
26 smaller amplitude values. Each register has inputs for signals A, B and C, and  
27 corresponding outputs.

28 Connected to each register is a multiplexer, such as multiplexers 102a through  
29 102c shown in Fig. 8. Each multiplexer is controlled by a comparator, such as  
30 comparators 101a through 101c shown in Fig. 8. Each comparator compares the current  
31 final sum A with the final sum R stored in the register connected to the comparator. The

1 comparator outputs a control signal, such as signals  $G_1$  through  $G_3$ , shown in Fig. 8, based  
2 on the comparison that controls the multiplexer connected to the outputs of that register.  
3 For example, comparator 101a compares the current final sum A with the final sum R  
4 stored in register 100a. If R is less than or equal to A, then comparator 101a outputs a  
5 control signal  $G_1$  to control multiplexer 102a.

6 The maximum finder operates by comparing the amplitude of the current final  
7 sum signal input from the matched filter to the amplitudes of previously received final  
8 sums that have been stored in all of the registers of the maximum finder. If the amplitude  
9 of the input signal is greater than or equal to the amplitudes recorded in some of the  
10 registers, then those registers are shifted downward, as arranged in Fig. 8, and the value  
11 of the input signal's amplitude is written to the register corresponding to the highest  
12 position among those shifted values. An amplitude of zero in a register marks it as  
13 empty. This convention allows the sorting process to start with empty registers and to  
14 continue properly when not all of the registers are full.

15 If partial or incremental dumping is to be permitted, then the lowest registers to  
16 be dumped, as illustrated in Fig. 8, that correspond to the registers holding the smallest  
17 amplitude values to be dumped, need to be dumped first to avoid the creation of 'holes'  
18 in the middle of the data. The logic for dumping those registers is not shown in Fig. 8.  
19 However, the preferred policy for maximum performance is to choose the best data: that  
20 is, to dump from the highest register, shifting the data upward. This policy ensures that  
21 when the processing algorithm receiving the output from the maximum finder, such as  
22 time/range detector 43, cannot take data from the maximum finder as fast as it is  
23 collected, the worst data is abandoned, rather than the best.

24 Referring to the logic diagram of Fig. 8 and using +1 to represent an upward shift,  
25 -1 for a downward shift, and 0 for no shift, a typical shift pattern for an insertion into a  
26 register of the maximum finder would be 0,0,-1,-1, and the shift pattern for dumping  
27 would be +1,+1,+1,+1, and 0,0,0,0 for no dumping. These conflicting shifts could be  
28 done in alternate clock cycles if the clock rate were doubled, such as to twice the chip  
29 rate, but this could increase the clock rate to an undesirable degree. An alternative  
30 method is to accommodate both data movements in the same clock cycle. Adding the  
31 two example shift patterns (dumping while inserting) yields the pattern +1,+1,0,0. This

1 pattern can be handled by designing the logic to operate like a left/right shift register.  
2 Each stage of such a register has a three-way multiplexer for selecting data from its left  
3 neighbor, right neighbor, or from the input signals.

4 The logic diagram of Fig. 8 is described next. Here, the registers 100a, 100b,  
5 100c, etc. are referred to as being sequentially numbered 1,2,3,..., starting from the top  
6 register 101a in Fig. 8, i.e., the register holding the maximum value.

7 Register  $n$  is referred to as  $R_n$ , and  $RA_n$  is the portion of the register holding the  
8 value "A."  $D$  is the dump enable signal. Comparator output  $G_n$  is true when  $RA_n \leq A$ .  
9 Accordingly, the clock should be enabled for register  $n$  when the condition  $\{D \text{ xor } G_n\}$   
10 is true, and the multiplexer inputs for register  $n$  should be enabled according to the  
11 following rules:

- 12 1) Enable input from  $R_{n-1}$  when  $\{\text{not } D \text{ and } G_{n-1}\}$  is true.
- 13 2) Enable input from A,B,C when  $\{\text{not } D \text{ and not } G_{n-1} \text{ or } D \text{ and } G_{n+1}\}$  is true.
- 14 3) Enable input from  $R_{n+1}$  when  $\{D \text{ and not } G_{n+1}\}$  is true.

15 These iterated rules are terminated at the top and bottom as follows:

- 16 1) At the top register, substitute A,B,C for  $RA_{n-1}$  and  $\{\text{false}\}$  for  $G_{n-1}$ . Also,  
17 take the dump output directly from the input A,B,C when  $G_1$  is true, else from  $R_1$ .
- 18 2) At the bottom register, substitute the pattern 0,X,X for  $RA_{n+1}$  and  $\{\text{false}\}$  for  
19  $G_{n+1}$ , where "X" means don't-care data.

20 Fig. 9 is a diagram showing how two correlation sums, X and Y are input into a  
21 maximum finder ranking list. The maximum finder is conceptually illustrated here as a  
22 list 103. In the list are stored correlation sums,  $A_1$  through  $A_6$ . These sums can be stored  
23 in registers 100a, 100b, 100c, etc. of the maximum finder shown in Fig. 8. In a first  
24 instance, a correlation sum "X" is within the range of amplitude values present in the list.  
25 Here, for example, X greater than  $A_4$ , but less than  $A_3$ . Accordingly, the maximum finder  
26 of Fig. 8 operates to insert X in the list between  $A_4$  and  $A_3$ , and would shift  $A_4$  through  
27  $A_6$  down one level in the maximum finder, assuming the list is not full. If the list is full,  
28  $A_6$  is discarded,  $A_5$  is shifted into the register occupied by  $A_6$ , and  $A_4$  is shifted into the  
29 register occupied by  $A_5$ , and X is stored in the register occupied by  $A_4$ .  $A_1$  through  $A_3$  are  
30 not shifted.

31 In the second instance, correlation sum Y is less than  $A_6$ . If the maximum finder

